OIPE IAP69 OCT 03 2006 PATENT & TRADEMARK OFFICE

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of	
KINZHALIN et al.	Examiner: Steelman, Mary J.
Application No: 09/881,791	Art Unit: 2191
Filed: June 14, 2001	Docket No.: SUNMP016
For: SYSTEM AND METHOD FOR AUTOMATED ASSERTION ACQUISITION IN A JAVA COMPATIBILITY TESTING ENVIRONMENT	Date: September 27, 2006

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on **September 27, 2006.**

Signed: _____
Kenneth D. Wright

TRANSMITTAL OF REPLY BRIEF
(PATENT APPLICATION -- 37 CFR 1.193)

Mail Stop: Appeal Brief-Patents
Commissioner for Patents
Alexandria, VA 22313-1450

Sir:

This Reply Brief is in response to the Examiner's Answer mailed July 27, 2006. The due date for this Reply Brief is September 27, 2006.

Applicants believe that no fees are due in connection with the filing of this Reply Brief. However, the Commissioner is authorized to charge any requied fees unknown to the Applicants to Deposit Account No. 50-0850, (Order No. SUNMP016). One additional copy of this transmittal is enclosed for potential fee processing.

Respectfully submitted,
MARTINE PENILLA & GENCARELLA, LLP

Kenneth D. Wright
Reg. No. 53,795

710 Lakeway Drive, Suite 200
Sunnyvale, CA 94085
(408) 749-6900
Customer No. 32291

OIPE IAP69 OCT 03 2006 PATENT & TRADEMARK OFFICE

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of	
KINZHALIN et al.	Examiner: Steelman, Mary J.
Application No: 09/881,791	Art Unit: 2191
Filed: June 14, 2001	Docket No.: SUNMP016
For: SYSTEM AND METHOD FOR AUTOMATED ASSERTION ACQUISITION IN A JAVA COMPATIBILITY TESTING ENVIRONMENT	Date: September 27, 2006

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on September 27, 2006.

Signed: _____
Kenneth D. Wright

REPLY BRIEF

Mail Stop: Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This Reply Brief is in response to the Examiner's Answer dated July 27, 2006. This Reply Brief is filed within the two-month time period extending to September 27, 2006. Please enter the following remarks.

The Listing of Claims on Appeal begins on page 2 of this Reply Brief.

Remarks/Arguments begin on page 6 of this Reply Brief.

LISTING OF CLAIMS ON APPEAL

1. A method for automated acquisition of assertions in a specification of a computer program, comprising:

receiving the specification as an input, wherein the specification includes a plurality of sentences describing the computer program;

obtaining a sentence from the plurality of sentences;

determining whether the obtained sentence is a testable assertion, wherein the testable assertion describes behavior of an application programming interface that can be tested;

marking the obtained sentence as testable when the obtained sentence is a testable assertion; and

using the sentences marked as testable to determine whether a test suite for testing the computer program is adequate.

2. The method as recited in claim 1, further comprising:

identifying a context within the specification.

3. The method as recited in claim 2, wherein the operation of obtaining the sentence from the plurality of sentences includes parsing the context to obtain the sentence.

4. The method as recited in claim 3, further comprising:

adding the marked obtained sentence to an assertion result set.

5. The method as recited in claim 4, wherein the context is a set of circumstances related to the obtained sentence.

6. The method as recited in claim 5, wherein each assertion includes at least one sentence of the specification.

7. The method as recited in claim 5, wherein each assertion includes at least two sentences of the specification.

8. A computer readable media including program instructions for automatically obtaining assertions from a specification of a computer program, comprising:

a code segment that receives the specification as an input;

a code segment that identifies a context within the specification;

a code segment that parses the identified context to obtain sentences;

a code segment that determines whether the obtained sentences are testable assertions, wherein each testable assertion is a sentence that describes behavior of an application programming interface that can be tested; and

a code segment that adds the testable assertions to an assertion result set, wherein the assertion result set can be used to facilitate testing of the specification.

9. The computer readable media of claim 8, further comprising:

a code segment that filters the identified context prior to parsing the context.

10. The computer readable media of claim 9, wherein the code segment that receives the specification is defined to receive the specification in a text format.

11. The computer readable media of claim 9, wherein the context is a set of circumstances related to the obtained sentences.

12. The computer readable media of claim 9, wherein each assertion includes at least one sentence of the specification.

13. The computer readable media of claim 9, wherein each assertion includes at least two sentences of the specification.

14. A computer readable media including program instructions for automated acquisition of assertions in a specification of a computer program, comprising:

a code segment that receives the specification in a text format, wherein the specification includes a plurality of sentences;

a code segment that obtains a sentence from the plurality of sentences;

a code segment that determines whether the obtained sentence is a testable assertion, wherein the testable assertion describes behavior of an application programming interface that can be tested; and

a code segment that marks the obtained sentence as testable when the obtained sentence is a testable assertion.

15. The computer readable media of claim 14, further comprising:

a code segment that identifies a context within the specification.

16. The computer readable media of claim 15, wherein the code segment that obtains the sentence from the plurality of sentences includes a code segment that parses the context to obtain the sentence.

17. The computer readable media of claim 16, further comprising:

a code segment that adds the marked obtained sentence to an assertion result set.

18. The computer readable media of claim 17, wherein the context is a set of circumstances related to the obtained sentence.

19. The computer readable media of claim 18, wherein each assertion includes at least one sentence of the specification.

20. The computer readable media of claim 19, wherein each assertion includes at least two sentences of the specification.

REMARKS/ARGUMENTS

This Reply Brief is in response to the Examiner's Answer dated July 27, 2006. This Reply Brief is filed within the two-month time period extending to September 27, 2006.

Response to Examiner's Answer

The Examiner continues to assert the same grounds of rejection addressed in the Appeal Brief filed April 18, 2006. Therefore, the Applicant's arguments presented in the Appeal Brief continue to be argued against the rejections of the claims on appeal. In the interest of brevity, the Applicant's respectfully request the Board of Patent Appeals and Interferences (the "Board") to refer to the Applicant's Appeal Brief of April 18, 2006, for a full explanation of the Applicant's position with respect to the Examiner's rejections. The remainder of the present Reply Brief responds specifically to the Examiner's comments as provided in the "Response to Argument" section of the Examiner's Answer.

Claim 1 recites a method for automated acquisition of assertions in a specification of a computer program. Claim 1 recites an operation for receiving the specification as an input. Claim 1 recites that the specification includes a plurality of sentences describing the computer program.

The Examiner has incorrectly asserted that the specification (of the application) fails to explicitly define "a specification of a computer program." The specification (p. 6, lines 5-8) states the following:

> "In one embodiment, a method for automated acquisition of assertions in a specification of a computer program is disclosed. An input specification is received, wherein the input specification comprises a plurality of sentences."

The specification (p.22, line 1, through p. 23, line 18) describes a process for obtaining specification assertions and validating the assertions with regard to Figure 5B. In this process, an operation 550 is performed to determine whether a sentence is available in

the specification of the computer program (p. 22, lines 16-20). Then, in an operation 554, a decision is made as to whether the obtained sentence is a testable assertion (p. 22, line 21-22). The specification (p. 22, line 22, through p. 23, line 10) further states the following:

> "In one embodiment, a natural language processing system can be used to process the obtained sentence. In this case, the natural language processing system includes an input means for inputting the sentence obtained from the specification, and a knowledge base for storing linguistic knowledge and general knowledge. In addition, a partitioner is included that partitions the sentence into words, and a derivation module is included that refers to knowledge stored in the knowledge base and derives concepts respectively represented by the words obtained by the partitioner. Further, an integration module can be included that relates the concepts of the words, which are derived by the derivation module, with one another by referring to knowledge stored in the knowledge base. For example, a valid assertion can be identified as a sentence which uses particular keywords or phrases such as 'required to' 'should', 'should not'."

At least in view of the foregoing, the Applicants submit that the specification does in fact explicitly define "a specification of a computer program," as recited in claim 1. In particularly, the specification explicitly defines the "specification of the computer program" as including a plurality of sentences describing the computer program. This fact is evidenced even further by reference to the exemplary portion of the computer program specification shown in Table 1 (p. 20), and the list of assertions extracted therefrom in the form of sentences as shown in Table 2 (p. 21).

The Examiner states that "Table 1 fails to convince the Examiner that a specification of a computer program must be a conversant textual description." The Examiner must consider the invention as being defined by that which is recited in the claim. Specifically with regard to claim 1, the Examiner must consider the recited specification of the computer program as including a plurality of sentences describing the computer program. Furthermore, as discussed above, the specification discloses that the plurality of sentences which describe the computer program represent conversant text. To establish prima facie obviousness of a claimed invention, all the claim limitations must be

taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). Furthermore, "All words in a claim must be considered in judging the patentability of that claim against the prior art." In re Wilson, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970). It is evident that the Examiner is not considering the recited feature of claim 1 which states that the specification of the computer program includes a plurality of sentences describing the computer program.

With regard to claim 1, the Examiner continues to assert that source file of Pavela (column 2, lines 12-17) teaches receiving a specification of a computer program, wherein the specification includes a plurality of sentences describing the computer program. It should be understood that the source file of Pavela is not disclosed as including a plurality of sentences describing a computer program. The Examiner has also referred to Figure 6, Item 602, of Pavela as teaching a code segment for an input specification. First, the source file template shown in Figure 6 of Pavela is not a specification of a computer program, wherein the specification includes a plurality of sentences describing the computer program, as recited in claim 1. Second, Item 602 in Figure 6 of Pavela is not a sentence.

Further with regard to claim 1, the Examiner continues to incorrectly assert that Pavela teaches the recited operation of determining whether the obtained sentence is a testable assertion, wherein the testable assertion describes behavior of an application programming interface that can be tested. Because the combination of Pavela and MPCD fails to teach receiving a specification of a computer program as an input, wherein the specification includes a plurality of sentences describing the computer program, and obtaining a sentence from the plurality of sentences, it follows that the combination of Pavela and MPCD cannot possible teach determining whether the obtained sentence is a testable assertion.

The Examiner states that Pavela (column 2, lines 12-17, and column 6, lines 28-30) "clearly" discloses determining whether the sentence obtained from the specification of the computer program is a testable assertion, wherein the testable assertion describes behavior of program that can be tested. The Examiner is simply wrong.

Pavela (column 2, lines 12-17) states the following:

> "The method comprises the steps of defining a source file having a plurality of tags associated with a member of a library of executable code objects defining a set of instructions for performing a portion of the automatic test procedure, generating a test plan in a conventional language from the source file, and generating an automated test code for the automated test procedure from the source file."

Pavela (column 6, lines 28-30) states the following:

> "Other tags, such as tags 602A-616A, are instead associated with the members of the library of executable test code objects 314."

As discussed previously, the source file of Pavela does not teach the specification of a computer program as recited in claim 1. Also, the source file of Pavela does not include sentences describing the computer program. It should be understood that the tags present within the source file of Pavela are not sentences describing the computer program. Also, Pavela does not teach obtaining a sentence from a specification of a computer program. Also, Pavela does not teach determining whether a sentence obtained from a specification of the computer program is a testable assertion. Moreover, Pavela does not teach an operation for determining whether anything is a testable assertion.

The foregoing notwithstanding, the tags present in the source file of Pavela are specifically defined to refer an object representing a set of instructions for performing a test procedure. Therefore, Pavela already knows that the tags refer to a set of instructions for performing a test procedure. Thus, Pavela does not teach or suggest an operation for determining whether the tags, or anything else, is a testable assertion. Contrary to the Examiner's position, the tags of Pavela do not represent sentences, much less sentences that

are testable assertions, much less wherein the testable assertions describe the behavior of a program that can be tested.

The Examiner continues to accuse the Applicants of attacking the Pavela and MPCD references individually to show non-obviousness. The Examiner has only relied upon MPCD to provide a definition of an application programming interface. The Examiner has used the definition of the application programming interface, as provided by the MPCD reference, to interpret the claimed term "application programming interface" to mean a "program." This effort by the Examiner to interpret the claimed feature of "application programming interface" as "program" is made in an attempt to force the teachings of Pavela on claim 1. However, even if the "application programming interface" of claim 1 is interpreted as a "program", the combined teachings of Pavela and MPCD still fail to teach each and every feature of claim 1, as required to establish a case of prima facie obviousness. The Applicants have not inappropriately attacked the cited references on an individual basis. Rather, the Applicants have addressed the cited reference teachings as they have been asserted by the Examiner.

In further regard to claim 1, Examiner continues to assert that generation of the test index as taught by Pavela is equivalent to the recited operation of marking the obtained sentence as testable when the obtained sentence is a testable assertion. The test index generated in Pavela simply represents a listing of system elements that are tested by a test case, wherein the test case is defined manually using the source file as an input vehicle. Again, because Pavela does not teach obtaining sentences from the specification of a computer program and determining whether the obtained sentences represent a testable assertion, it is not reasonable to conclude that Pavela teaches anything associated with such sentences, particularly marking the obtained sentences as testable when the obtained sentence represents a testable assertion.

It should be appreciated that each of independent claims 8 and 14 recite features similar to those discussed above with regard to claim 1. Therefore, the Applicants' arguments presented above with regard to claim 1 are equally applicable to the similar features recited in claims 8 and 14. Further with respect to claim 8, Pavela does not teach identification of a context within the specification of the computer program. Additionally, Pavela does not disclose parsing the context within the specification of the computer program to obtain sentences. The teachings of Pavela as cited by the Examiner with regard to above-mentioned features of claim 8 are simply not relevant.

To establish prima facie obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). For at least the reasons discussed above and in the Appeal Brief of April 18, 2006, the Applicants submit that the combined references fail to teach or suggest each and every feature of claims 1-20, respectively. Therefore, the Board is respectfully requested to overturn the Examiner's rejections of claim 1-20 under 35 U.S.C. 103.

If the Examiner has any questions concerning the present Reply Brief, the Examiner is requested to contact the undersigned at (408) 774-6914. If any other fees are due in connection with filing this Reply Brief, the Commissioner is also authorized to charge Deposit Account No. 50-0805 (Order No. SUNMP016). A duplicate copy of the transmittal is enclosed for this purpose.

Respectfully submitted,
MARTINE PENILLA & GENCARELLA, LLP

Kenneth D. Wright
Reg. No. 53,795

710 Lakeway Drive, Suite 200
Sunnyvale, CA 94085
Telephone: (408) 749-6900
Customer Number 32291